IN THE CLAIMS

Please amend the claims as follows:

1-2. (canceled)

3. (previously presented) The method of claim 14, wherein storing variations is
    performed using a pointer.

4. (previously presented) The method of claim 14, further comprising, before presenting
    the represented sequence to a user:
    identifying, replets that can be used to represent multiple subsequences.

5. (previously presented) The method of claim 14, further comprising, before presenting
    the reconstructed represented sequence to a user:
    segmenting the matching subsequences into multiple parts to account for
    location-specific variations of the matching subsequences in the sequence data.

6. (previously presented) The method of claim 14, further comprising, before presenting
the reconstructed represented sequence to a user:
    storing replet information in a replet-information table using a pointer, so that
equivalent replet sequences occupy single storage space.

7- 9. (canceled)

10. (previously presented) The method of claim 14, further comprising, before
presenting the reconstructed represented sequence to a user:
    storing multiple views of the sequence data at multiple levels of abstraction.

11-13. (canceled)

Docket JP920030152US1

Appl. No.: 10/699,024
Filing Date: October 31, 2003

14. (previously presented) A computer system-implemented method for storing and presenting sequence data, comprising:

i) specifying a set of replets for analysis by a computer system;

ii) for each replet in the set, comparing each replet by the computer system to a sequence for determining by the computer system a subsequence of the sequence that matches each replet, if any;

iii) generating entries of match-set data structures by the computer system responsive to the comparing, one match-set data structure for each replet, a match-set data structure comprising a sequence identification to identify a subsequence where a match occurs and offset information to determine a position within the sequence where the matching subsequence of the sequence is located, wherein the offset information comprises a first and second position parameter, the first position parameter denoting a location in the sequence and the second position parameter denoting an offset from the location;

iv) storing the generated entries of the match-set data structures in a computer readable memory by the computer system;

v) deleting by the computer system each matching subsequence from the sequence where it is found;

vi) concatenating by the computer system, in order, unmatched regions of sequences that remain after deleting each matching subsequence to form and store a backbone sequence;

vii) the computer system generating a first representation of the sequence responsive to the stored backbone sequence and at least a portion of the stored match-set data and presenting the represented sequence to a user of the computer system;

wherein one of the subsequences is matched by a certain plurality of the replets and the method further includes:

viii) the computer system generating or receiving a selection of one of the certain plurality of replets, wherein generating the first representation in step vii) is responsive to the selected one of the certain plurality of replets but not responsive to any non-selected replet, and wherein the selected one of the replets has a position

Docket JP920030152US1

Appl. No.: 10/699,024
Filing Date: October 31, 2003

within the sequence, and wherein after receiving the selection the computer system updates the first and second parameters for any non-selected replet associated with the selected one of the replets, the updating being responsive to the position of the selected one of the replets;

ix) the computer system generating or receiving a selection of a replet that was not selected in step viii); and

x) the computer system generating and presenting a second representation of the sequence, wherein the representation in step x) is responsive to the one of the replets indicated by the selection in step ix) instead of the one of the replets indicated by the selection in step viii) and wherein the computer system performs the generating in step x) by reference to the first and second parameters for the one of the replets indicated by the selection in step ix), such that the computer system performs the generating in step x) using the position parameters updated in step viii).

15. (previously presented) The method of claim 14, wherein, prior to presenting the represented sequence to the user, variations are stored in a list data structure comprising a variation identification.

16. (previously presented) The method of claim 15, wherein the list data structure comprises a subsequence character that matches a "don't care" character in a replet that matches the subsequence.

17. (previously presented) The method of claim 16, wherein, prior to presenting the represented sequence to the user, the position of the subsequence character within the subsequence is stored in the list data structure.

18. (previously presented) The method of claim 15, wherein an indirection pointer points to a variation so that variations common to more than one subsequence are not stored more than once.

19. (previously presented) The method of claim 14, wherein a represented sequence presented to the user is in response to a query by the user.

20. (previously presented) The method of claim 19, wherein a query specifies a replet.

21. (previously presented) The method of claim 14, wherein specifying a set of one or more replets comprises adding a new replet to a pre-existing set for which the steps of claim 14 have been performed.

22. (previously presented) The method of claim 14, wherein specifying a set of one or more replets comprises removing a replet from a pre-existing set for which the steps of claim 14 have been performed.

23. (canceled)

24. (new) A computer program product for storing and presenting sequence data, the computer program product including a computer readable storage medium having instructions stored thereon for execution by a computer, wherein the instructions, when executed by the computer, cause the computer to implement a method comprising the steps of:

    i) specifying a set of replets for analysis by a computer system;

    ii) for each replet in the set, comparing each replet by the computer system to a sequence for determining by the computer system a subsequence of the sequence that matches each replet, if any;

    iii) generating entries of match-set data structures by the computer system responsive to the comparing, one match-set data structure for each replet, a match-set data structure comprising a sequence identification to identify a subsequence where a match occurs and offset information to determine a position within the sequence where the matching subsequence of the sequence is located, wherein the offset information comprises a first and second position parameter, the first position parameter denoting a

Docket JP920030152US1

Appl. No.: 10/699,024
Filing Date: October 31, 2003

location in the sequence and the second position parameter denoting an offset from the
location;

iv) storing the generated entries of the match-set data structures in a computer
readable memory by the computer system;

v) deleting by the computer system each matching subsequence from the
sequence where it is found;

vi) concatenating by the computer system, in order, unmatched regions of
sequences that remain after deleting each matching subsequence to form and store a
backbone sequence;

vii) the computer system generating a first representation of the sequence
responsive to the stored backbone sequence and at least a portion of the stored match-
set data and presenting the represented sequence to a user of the computer system;

wherein one of the subsequences is matched by a certain plurality of the replets
and the method further includes:

viii) the computer system generating or receiving a selection of one of the
certain plurality of replets, wherein generating the first representation in step vii) is
responsive to the selected one of the certain plurality of replets but not responsive to
any non-selected replet, and wherein the selected one of the replets has a position
within the sequence, and wherein after receiving the selection the computer system
updates the first and second parameters for any non-selected replet associated with the
selected one of the replets, the updating being responsive to the position of the selected
one of the replets;

ix) the computer system generating or receiving a selection of a replet that was
not selected in step viii); and

x) the computer system generating and presenting a second representation of
the sequence, wherein the representation in step x) is responsive to the one of the
replets indicated by the selection in step ix) instead of the one of the replets indicated by
the selection in step viii) and wherein the computer system performs the generating in
step x) by reference to the first and second parameters for the one of the replets

Docket JP920030152US1

Appl. No.: 10/699,024
Filing Date: October 31, 2003

indicated by the selection in step ix), such that the computer system performs the generating in step x) using the position parameters updated in step viii).

25. (new) The computer program product of claim 24, wherein storing variations is performed using a pointer.

26. (new) The computer program product of claim 24, further comprising, before
    presenting the represented sequence to a user:
    identifying, replets that can be used to represent multiple subsequences.

27. (new) A computer system comprising:
    a processor; and
    a storage device connected to the processor, wherein the storage device has stored thereon a program for controlling the processor, and wherein the processor is operative with the program to execute the program for performing the steps of:
    i) specifying a set of replets for analysis by a computer system;
    ii) for each replet in the set, comparing each replet by the computer system to a sequence for determining by the computer system a subsequence of the sequence that matches each replet, if any;
    iii) generating entries of match-set data structures by the computer system responsive to the comparing, one match-set data structure for each replet, a match-set data structure comprising a sequence identification to identify a subsequence where a match occurs and offset information to determine a position within the sequence where the matching subsequence of the sequence is located, wherein the offset information comprises a first and second position parameter, the first position parameter denoting a location in the sequence and the second position parameter denoting an offset from the location;
    iv) storing the generated entries of the match-set data structures in a computer readable memory by the computer system;

Docket JP920030152US1

Appl. No.: 10/699,024
Filing Date: October 31, 2003

v) deleting by the computer system each matching subsequence from the sequence where it is found;

vi) concatenating by the computer system, in order, unmatched regions of sequences that remain after deleting each matching subsequence to form and store a backbone sequence;

vii) the computer system generating a first representation of the sequence responsive to the stored backbone sequence and at least a portion of the stored match-set data and presenting the represented sequence to a user of the computer system;

wherein one of the subsequences is matched by a certain plurality of the replets and the method further includes:

viii) the computer system generating or receiving a selection of one of the certain plurality of replets, wherein generating the first representation in step vii) is responsive to the selected one of the certain plurality of replets but not responsive to any non-selected replet, and wherein the selected one of the replets has a position within the sequence, and wherein after receiving the selection the computer system updates the first and second parameters for any non-selected replet associated with the selected one of the replets, the updating being responsive to the position of the selected one of the replets;

ix) the computer system generating or receiving a selection of a replet that was not selected in step viii); and

x) the computer system generating and presenting a second representation of the sequence, wherein the representation in step x) is responsive to the one of the replets indicated by the selection in step ix) instead of the one of the replets indicated by the selection in step viii) and wherein the computer system performs the generating in step x) by reference to the first and second parameters for the one of the replets indicated by the selection in step ix), such that the computer system performs the generating in step x) using the position parameters updated in step viii).

28. (new) The computer system of claim 27, wherein storing variations is performed using a pointer.